

通過テスト2 解答



2 クラスの仕組み

1. 問1、問2

```
package e1;

class Timer {
    int interval;
    int num;
    public Timer(int t1, int t2){
        interval = t1;
        num = t2;
    }
    public void show(){
        System.out.println("interval = " + interval);
        System.out.println("num = " + num);
    }
}
```

【解説】 コンストラクタはクラスと同名で戻り値がないことに注意する。フィールド変数に値を設定するのは主にコンストラクタの役割である。初めて値を設定するのでオブジェクトを初期化するともいう。インスタンスメソッドはフィールド変数を自由に使ってプログラミングできる。

問3

```
package e1;

public class UseTimer {
    public static void main(String[] args) {
        Timer t = new Timer(15, 5);
        t.show();
    }
}
```

【解説】 まず new でオブジェクトを作成するのが第一。つぎにメンバ参照演算子を使ってオブジェクトのメソッドを実行する。

2. 問1 String[] c

【解説】 コンストラクタの引数はフィールド変数を初期化するのに使用することから推理する.

配列変数も使えることに注意. 配列なので, C は参照型の変数で **String** 配列への参照が入っている.

問2

```
package e2;

public class UseHand {
    public static void main(String[] args) {
        String[] s1 = {"ハート 10", "スペード 2"};
        String[] s2 = {"ダイヤ 12", "ハート 5", "クラブ 2"};
        Hand h1 = new Hand(s1);
        Hand h2 = new Hand(s2);
    }
}
```

【解説】 配列を作成してそれを引数にしてオブジェクトを作成する. コンストラクタの引数の型は **String[]** であるから, 同じ型を使用することに注意. **h1, h2** には **String** 型配列への参照が入っている.

3. D

【解説】 C はコンストラクタ名が **dog** になっていて誤り.

4. B

【解説】 9 行目にあるのは引数のないコンストラクタだが, **Glass** クラスでは定義されていない.

5. D, E

【解説】 7 行目で作成した変数 **tom** は, 宣言しただけなので空. 空の変数にはオブジェクトの参照がセットされていないので, メンバ参照演算子をつかってオブジェクトを操作することはできない.

6. I

【解説】 **p2** には **null** がセットされている. **null** は有効ではない参照なのでメンバ参照演算子をつかってオブジェクトを操作することはできない. コンパイルエラーではなく実行時例外になる.